

## Exhibit A

Contents of file index.html:

```
<html>
<head>
<title>Loading New Search Engine: Copyright 2002 Alan Swahn. All Rights Reserved.</title>
<META NAME="copyright" CONTENT="Copyright 2002 Alan Swahn. All Rights Reserved.">
<META HTTP-EQUIV="Window-target" CONTENT=" _top">
<META NAME="keywords" CONTENT="Website Based Search Engine">
<META NAME="description" CONTENT="New Ideas in Searcing the Internet">
<Meta HTTP-EQUIV="content-type" content="text/html; charset=windows-1252">
<META HTTP-EQUIV="content-language" CONTENT="en-US">
<META NAME="author" CONTENT="Alan Swahn">
<script type="text/javascript" src="engine.js"></script>
</head>

<body scroll='no' onLoad='onloadpage()' onError='onerrorpage()' onAbort='onabortpage()' onBeforeUnload='onbeforeunloadpage()' onUnload='onunloadpage()'>

<IFRAME id='g1' name='g1' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:1; zoom:1.0;'></IFRAME>
<IFRAME id='g2' name='g2' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:2; zoom:1.0;'></IFRAME>
<IFRAME id='g3' name='g3' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:3; zoom:1.0;'></IFRAME>
<IFRAME id='g4' name='g4' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:4; zoom:1.0;'></IFRAME>
<IFRAME id='g5' name='g5' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:5; zoom:1.0;'></IFRAME>
<IFRAME id='g6' name='g6' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:6; zoom:1.0;'></IFRAME>
<IFRAME id='g7' name='g7' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:7; zoom:1.0;'></IFRAME>
<IFRAME id='g8' name='g8' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:8; zoom:1.0;'></IFRAME>
```





```

<!FRAME id='g44' name='g44' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:44; zoom:1.0;'></!FRAME>
<!FRAME id='g45' name='g45' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:45; zoom:1.0;'></!FRAME>
<!FRAME id='g46' name='g46' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:46; zoom:1.0;'></!FRAME>
<!FRAME id='g47' name='g47' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:47; zoom:1.0;'></!FRAME>
<!FRAME id='g48' name='g48' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:48; zoom:1.0;'></!FRAME>
<!FRAME id='g49' name='g49' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:49; zoom:1.0;'></!FRAME>
<!FRAME id='g50' name='g50' SECURITY='unrestricted' HSPACE='0' VSPACE='0' ALIGN='top' SCROLLING='auto' FRAMEBORDER='0' ALLOWTRANSPARENCY='false' STYLE='position:absolute; left:0; top:4%; width:102%; height:96%; z-index:50; zoom:1.0;'></!FRAME>

<DIV id='control' name='control' HSPACE='0' VSPACE='0' ALIGN='left' STYLE='background:navy; position: absolute; left:0; top:0; width:102%; height:4%; z-index:52;'>

<button id='btst'  STYLE='background:#000000; width:5%;' TITLE="Test on/off"          onclick="test()">Z</button>

<button id='badd2fav'  STYLE='background:#00FF00; width:4%;' TITLE="Add to IE Favorites"    onclick="add2favorites()" ondblclick="add2favorites()">Add</button>
<button id='bjumpurl'  STYLE='background:#00FF00; width:4%;' TITLE="Jump to Website"      onclick="jumpurl()" ondblclick="jumpurl()">Jmp</button>
<button id='bzoomup'   STYLE='background:#00FF00; width:4%;' TITLE="Zoom Page by 120%"    onclick="zoomup()" ondblclick="zoomup()">Up</button>
<button id='bzoomdown'  STYLE='background:#00FF00; width:4%;' TITLE="Zoom Page by -120%"   onclick="zoomdown()" ondblclick="zoomdown()">Dn</button>
<button id='bndisplay'  STYLE='background:#00FF00; width:4%;' TITLE="Multiple Display Select" onclick="setndisplay()" ondblclick="setndisplay()">nDsp</button>
<button id='bfnd'  STYLE='background:#00FF00; width:4%;' TITLE="Find WebSites"      onclick="findtext()" ondblclick="findtext()">Find</button>

<INPUT id='isubject'  STYLE='position: absolute; left:33%; width:30%; background:#FFFFFF; TYPE='text' VALUE="" SIZE="" TITLE="WebSite Topic to Find">

```

```
<Select id="iln" name="iln" STYLE="position:absolute; left:63%; width:5%; background:#FFFFFF;"><option value='50' selected>50<option value='40'>40<option value='30'>30<option value='20'>20<option value='10'>10</Select>

<button id="bsearch" STYLE="position:absolute; left:68%; width:5%; background:#00FF00;" TITLE="Search for WebSites" onclick="getnewlinks()" ondblclick="getnewlinks()">Search</button>
<button id="bremove" STYLE="position:absolute; left:73%; width:7%; background:#00FF00;" TITLE="Remove WebSite from List" onclick="removelink()" ondblclick="removelink()">Remove</button>
<button id="bfirst" STYLE="position:absolute; left:80%; width:5%; background:#00FF00;" TITLE="Display First WebSite Found" onclick="first()" ondblclick="first()">First</button>
<button id="blast" STYLE="position:absolute; left:85%; width:5%; background:#00FF00;" TITLE="Display Last WebSite Found" onclick="last()" ondblclick="last()">Last </button>
<button id="bback" STYLE="position:absolute; left:90%; width:5%; background:#00FF00;" TITLE="Go Back 1 WebSite" onclick="back()" ondblclick="back()">Back </button>
<button id="bnext" STYLE="position:absolute; left:95%; width:5%; background:#00FF00;" TITLE="Go Forward 1 WebSite" onclick="next()" ondblclick="next()">Next </button>
<BGSOUND id="audio" SRC="" VOLUME=0 BALANCE=0 LOOP=0 HIDDEN=true>
</DIV>

<APPLET ID='getlinks' NAME='getlinks' ALIGN='right' CODE='engine.class' VSPACE=0 HSPACE=0 WIDTH=0 HEIGHT=0 STYLE="position:relative; z-index:0; visibility:hidden;" MAYSCRIPT></APPLET>
```

```
</body>
</html>
```

## Contents of file engine.js:

```
//Globals
var links = new Array(); //storage for web addresses
var linkcount = 0; //running link count
var maxlinkent = 0; //max links found on subject returned by engine.class
var elDArray = new
Array("gx","g1","g2","g3","g4","g5","g6","g7","g8","g9","g10","g11","g12","g13","g14","g15","g16","g17","g18","g19","g20","g21","g22","g23","g24","g25",
"g26","g27","g28","g29","g30","g31","g32","g33","g34","g35","g36","g37","g38","g39","g40","g41","g42","g43","g44","g45","g46","g47","g48","g49","g50");
//iFrame IDs
var try2loadlink = new Array();
var bottomofstack = 1;
var displayent = 1;
var maxifframecount = 50;
var testing = false;
var defaulttitle = "Website Based Search Engine version 6.0 by Alan Swahn";
//if (! eval("navigator.javaEnabled()")) {alert("Please Enable Java on Your Computer to Use This Program");}
window.moveTo(0,0);
window.resizeTo(screen.availWidth,screen.availHeight);
var n = (document.layers) ? true:false;
var ie = (document.all) ? true:false;
//if(! ie){alert("Internet Explorer 6.0 Required"); top.window.close();}
*****End Globals*****
//-----
//Redefine Events
//-----
var catch_onblur=true;
function swallowonblur(){ //uses attach, detach methods
  if(catch_onblur){
    if(testing){alert(window.event.srcElement.id + " onblur event trapped: " + window.event.type);}
    //colortrack("bmodal");
    return null;
  }
}
```

```
//-----
var catch_onerrorupdate=true;
function swallowonerrorupdate() { //uses attach, detach methods
    if(catch_onerrorupdate){
        if(testing){alert(window.event.srcElement.id + " onerrorupdate event trapped: " + window.event.type);}
        document.getElementById(eIDArray[linkcount]).focus();
        return null;
    }
}

//-----
var catch_onerror=true;
function swallowonerror() { //uses attach, detach methods
    if(catch_onerror){
        if(testing){alert("onerror event trapped");}
        document.getElementById(eIDArray[linkcount]).focus();
        return null;
    }
}
//End Redefine Events
//-----

//Redefine Methods
//-----

//oldshowModalDialog is an equivlent Object to showModalDialog
//oldshowModalDialog must be defined before showModalDialog is redefined
var oldshowModalDialog=showModalDialog;
var catch_showModalDialog=true;

function swallowshowModalDialog(xurl,xarguments,xfeatures){

    if(catch_showModalDialog){
        if(testing){alert("Caught showModalDialog Method: " + event.type);}


```

```

//colortrack('bmodal');
document.getElementById(eIDArray[linkcount]).focus();
return null; //equivalent to clicking X out
}
else{
    return oldshowModalDialog(xurl,xarguments,xfeatures); //pass arguments to original function
}
}//end swallowshowModalDialog

//a call to showModalDialog event will now call swallowshowModalDialog instead
eval('function showModalDialog(xurl,xarguments,xfeatures){return swallowshowModalDialog(xurl,xarguments,xfeatures);}');

//-----
//oldshowModelessDialog is an equivalent Object to showModelessDialog
//oldshowModelessDialog must be defined before showModelessDialog is redefined
var oldshowModelessDialog=showModelessDialog;
var catch_showModelessDialog=true;

function swallowshowModelessDialog(xurl,xarguments,xfeatures){

    if(catch_showModelessDialog){
        if(testing){alert("Caught showModelessDialog Method: " + event.type);}
        //colortrack('bmodeless');
        document.getElementById(eIDArray[linkcount]).focus();
        return null; //equivalent to clicking X out
    }
    else{
        return oldshowModelessDialog(xurl,xarguments,xfeatures); //pass arguments to original function
    }
}//end swallowshowModelessDialog

//a call to showModelessDialog will now call swallowshowModelessDialog instead
eval('function showModelessDialog(xurl,xarguments,xfeatures){return swallowshowModelessDialog(xurl,xarguments,xfeatures);}');


```

```
//-----
//oldshowHelp is an equivalent Object to showHelp
//oldshowHelp must be defined before showHelp is redefined
var oldshowHelp=showHelp;
var catch_showHelp=true;

function swallowshowHelp(xurl,xcontext){
    if(catch_showHelp){
        if(testing){alert("Caught showHelp Method: " + event.type);}
        //colortrack('bhelp');
        document.getElementById(cIDArray[linkcount]).focus(); //event handler attached to window
        window.event.returnValue=false;
        return; //no return value from this function
    }
    else{
        return oldshowHelp(xurl,xcontext);
    }
}//end swallowshowHelp

//a call to showHelp will now call swallowshowHelp instead
eval('function showHelp(xurl,xcontext){return swallowshowHelp(xurl,xcontext);}');

//-----
var oldcreatePopup=window.createPopup; //must be outside swallowcreatePopup()
var catch_createPopup=true;

function swallowcreatePopup(xarguments){
    alert("popup triggered");
    if(catch_createPopup){
        if(testing){alert("Caught createPopup Method: " + event.type);}
        //colortrack('bpopup');
        document.getElementById(cIDArray[linkcount]).focus(); //event handler attached to window
        return null;
    }
}
```

```
        }
    else{
        return oldcreatePopup(xarguments);
    }
}//end swallowcreatePopup

// a call to createPopup will now call swallowcreatePopup instead
eval('function createPopup(xarguments){return swallowcreatePopup(xarguments);}');

//-----
var oldConfirm=confirm;
var catch_confirm=true;

function swallowconfirm(xtext){
    if(catch_confirm){
        if(testing){alert("Caught confirm method: " + event.type);}
        //colortrack('bconfirm');
        document.getElementById(eIDArray[linkcount]).focus(); //event handler attached to window
        return null; //only true or false are valid returns
    }
    else{
        return oldConfirm(xtext);
    }
}//end swallowconfirm

// a call to Confirm will now call swallowconfirm instead
eval('function confirm(xtext){return swallowconfirm(xtext);}');

//-----
var oldPrompt=prompt;
var catch_prompt=true;
var returnstring = "",

function swallowprompt(xtext,defaultvalue){
```

```
if(catch_prompt){
    if(testing){alert("caught prompt method: " + event.type);}
    //colortrack('bprompt');
    document.getElementById(cIDArray[linkcount]).focus(); //event handler attached to window
    return returnstring; //string or integer of user response allowed as return values
}
else{
    return oldPrompt(xtext,defaultvalue);
}
}//end swallowprompt
```

```
//a call to Prompt will now call swallowprompt instead
eval(function prompt(xtext,defaultvalue){return swallowprompt(xtext,defaultvalue);});
```

```
//-----
```

```
var oldAlert=alert;
var catch_alert=false;
```

```
function swallowalert(xtext){
    if(catch_alert){
        if(testing){window.status="caught alert method: " + event.type;}
        //colortrack('balert');
        document.getElementById(cIDArray[linkcount]).focus(); //event handler attached to window
        return null;
    }
    else{
        return oldAlert(xtext);
    }
}//end swallowalert
```

```
//a call to Alert event will now call swallowalert instead
eval(function alert(xtext){return swallowalert(xtext);});
```

```
//-----
var oldOpen=open;
var catch_open=true;
var counter=0;

function xswallowopen(xurl,xtargetname,xfeatures,xreplace){
counter++;
    if(catch_open){
        window.status="caught open method count: " + counter;
        //colortrack('bopen');
        document.getElementById(eIDArray[linkcount]).focus(); //event handler attached to window
        return null;
    }
    else{
        return oldOpen(xurl,xtargetname,xfeatures,xreplace);
    }
}//end swallowopen

//a call to open will now call swallowopen instead
//eval('function open(xurl,xtargetname,xfeatures,xreplace){return swallowopen(xurl,xtargetname,xfeatures,xreplace);}');

//-----
var oldsetTimeout=setTimeout;
var catch_setTimeout=true;

function swallowsetTimeout(xcode,xmilliseconds,xlanguage){
    if(catch_setTimeout){
        if(testing){alert("caught setTimeout window event: " + event.type);}
        //colortrack('btimeout');
        document.getElementById(eIDArray[linkcount]).focus(); //event handler attached to window
        return 0; //integer identifier returned
    }
    else{

```

```

        return oldsetTimeout(xcode,xmilliseconds,xlanguage);
    }
}//end swallowsetTimeout

//a call to setTimeout will now call swallowsetTimeout instead
eval("function setTimeout(xcode,xmilliseconds,xlanguage){return swallowsetTimeout(xcode,xmilliseconds,xlanguage);}");
//-----

var oldsetInterval=setInterval;
var catch_setInterval=true;

function swallowsetInterval(xcode,xmilliseconds,xlanguage){
    if(catch_setInterval){
        if(testing){alert("caught setInterval window event: " + event.type);}
        //colortrack("binterval");
        document.getElementById(cIDArray[linkcount]).focus(); //event handler attached to window
        return 0; //integer identifier returned
    }
    else{
        return oldsetInterval(xcode,xmilliseconds,xlanguage);
    }
}//end swallowsetInterval

//a call to setInterval will now call swallowsetInterval instead
eval("function setInterval(xcode,xmilliseconds,xlanguage){return swallowsetInterval(xcode,xmilliseconds,xlanguage);}");
//-----
*****End Redefine Methods*****
//-----
//-----
//Function Definitions
//-----


function setlinkcount(maxlinkcount) { //called from engine to set number of links to be returned
    linkcount = 0; //initialize link counter
}

```

```

maxlinkent = maxlinkcount;
initializeiframes();
if(maxlinkent > maxiframecount){maxlinkent = maxiframecount;} //engine may return more links than can be displayed, therefore limit to what can be
displayed
}
//-----
function linker(fsrc) { //very dependent on engine actually returning all links associated with maxlinkent
    linkcount++;
    if(linkcount > maxlinkent){return true;} //engine is trying to provide more links than can be displayed; disregard such links
    links[linkcount] = fsrc; //index starts at 1; Load all links references returned from engine
    try2loadlink[linkcount] = false; //links are not actually loaded into iFrames yet

    if(linkcount == maxlinkent){ //all links references loaded
        linkcount=bottomofstack; //reset running link counter to bottom of stack
        loadcurrent(); //load first link into hidden frame
        loadnext(); //load following link
        displaycurrent(); //display first link
    }
}
//-----

function hidecurrent() {
    document.getElementById(eIDArray[linkcount]).style.visibility="hidden"; //make next iFrame visible
}

function displaycurrent() {
    if(linkcount <= maxlinkent){
        if(readystate()){//make next iFrame visible if already loaded
            setstatus(); //reset title and status bar
            document.getElementById(eIDArray[linkcount]).style.visibility="visible";
            document.getElementById(eIDArray[linkcount]).focus(); //setfocus to iFrame to fire event, ie new window will generate blur
        }
        else{
            loadcurrent(); //reload current frame
        }
    }
}

```

```

        setstatus(); //reset title and status bar
        document.getElementById(eIDArray[linkcount]).style.visibility="visible";
        document.getElementById(eIDArray[linkcount]).focus();
    }
}

}//end display current
//-----
function loadcurrent(){
    document.getElementById(eIDArray[linkcount]).src = links[linkcount]; //load next link into frame
    try2loadlink[linkcount] = true;
}
}//end loadcurrent
//-----

function loadnext(){
    if(linkcount < maxlinkcnt){ //array guard
        document.getElementById(eIDArray[linkcount+1]).src = links[linkcount+1]; //load link into hidden frame
        try2loadlink[linkcount+1] = true;
    }
}
}//end loadnext
//-----

function next(){
    if(maxlinkcnt == 0 && issubject.value == ""){return;}
    if(maxlinkcnt == 0 && issubject.value != ""){getnewlinks(); return;}
    playsound('buttonpush.wav');

    hidecurrent();
    linkcount++; //new display position
    if(linkcount<=maxlinkcnt){
        displaycurrent();
        if(linkcount < maxlinkcnt){loadnext();}
    }
    else{
        linkcount = maxlinkcnt; //set to last valid link
    }
}

```

```
    hidecurrent();      //hide current link
    linkcount = bottomofstack; //reset running link counter to beginning
    displaycurrent();   //display first link
}
}//end next
//-----
```

```
function back(){
    if(maxlinkcnt == 0){return;}
    playsound('buttonpush.wav');
    hidecurrent();
    linkcount--; //new display position

    if(linkcount>=bottomofstack){
        displaycurrent();
    }
    else{
        linkcount = maxlinkcnt; //set to last valid link
        displaycurrent(); //display first link
    }
}//end back
//-----
```

```
function first(){
    if(maxlinkcnt == 0){return;}
    playsound('buttonpush.wav');
    hidecurrent(); //hide current link
    linkcount=bottomofstack; //reset running link counter
    displaycurrent(); //display first link
}//end first
//-----
```

```
function last(){
    if(maxlinkcnt == 0){return;}
    playsound('buttonpush.wav');
```

```

    hidecurrent(); //hide current link
    linkcount = maxlinkcnt; //reset running link counter to maximum
    displaycurrent(); //display last link
}//end last
//-----

function attacheventhandlers(){
    var i;
    var rs1=false;
    var rs2=false;
    var rs3=false;
    for(i=bottomofstack; i <= maxiframecount; i++){
        rs1=document.getElementById(eIDArray[i]).attachEvent('onblur',swallowonblur);
        rs2=document.getElementById(eIDArray[i]).attachEvent('onerrorupdate',swallowonerrorupdate);
        rs3=window.attachEvent('onerror',swallowonerror);
        if(testing && (! rs1 || ! rs2 || ! rs3)){Alert("Error attaching events");}
    }
}//end attacheventhandlers
//-----

function detacheventhandlers(){
    var i;
    for(i=bottomofstack; i <= maxiframecount; i++){
        document.getElementById(eIDArray[i]).detachEvent('onblur',swallowonblur);
        document.getElementById(eIDArray[i]).detachEvent('onerrorupdate',swallowonerrorupdate);
        window.detachEvent('onerror',swallowonerror); //no return value from detach
    }
}//end detacheventhandlers
//-----

function initializeframes() //initialize and hide all iFrames
{
    var i;
    window.status="New Ideas on Searching the Internet";
    window.document.title=defaulttitle;
    for(i=bottomofstack; i <= maxiframecount; i++){

```

```
document.getElementById(eIDArray[i]).style.visibility='hidden'; //reset visibility on reload
document.getElementById(eIDArray[i]).style.zoom='1.0'; //reset zoom on reload
}
}//end initializeframes
//-----
function setstatus(){
    document.getElementById("bjumpurl").value = "J" + linkcount;
    window.status="New Ideas on Searching the Internet";
    window.document.title=links[linkcount];
}//end setstatus
//-----
function readystate(){
    //alert("readystate: " + document.getElementById(eIDArray[linkcount]).readyState);
    //alert("try2load: " + try2loadlink[linkcount]);
    if(document.getElementById(eIDArray[linkcount]).readyState == "complete" && try2loadlink[linkcount]){
        bnext.style.backgroundColor="#00ff00";
        bback.style.backgroundColor="#00ff00";
        return true;
    }else{
        bnext.style.backgroundColor="orange";
        bback.style.backgroundColor="orange";
        return false;
    }
}//end readystate
//-----
function getnewlinks(){
    if(issubject.value == ""){return;} //user send empty subject, return
    document.getlinks.setsubject(issubject.value); //document.APPLET.EngineProcedure.PassParameter
    document.getlinks.setlinkgetcount(iln.value);
    playsound('accessing.wav');
    document.getlinks.enginestart();
}//end getnewlinks
//-----
```

```
function jumpurl(){
    var url=window.document.title;
    if(url=="" | url==defaulttitle){return;}
    if(url.indexOf("http://") > -1){
        window.location.href = url;
    }
    else{
        window.location.href = "http://" + url;
    }
}//end jumpurl
//-----

function add2favorites(){
    if(window.document.title == ""){return;}
    window.external.AddFavorite(window.document.title, isubject.value);
}//end add2favorites
//-----

function playsound(xfile){
    document.getElementById('audio').src = xfile;
    //document.getElementById('audio').run();
}//end playsound
//-----

function removelink(){
    if(maxlinkcnt==1){return;} //if only 1 link left, don't remove
    for(i = linkcount; i < maxlinkcnt; i++){ //shift stack down 1 position
        document.getElementById(eIDArray[i]).visibility='hidden';
        links[i]=links[i+1]; //starting at link to be removed, drop stack by one position
    }
    //if at top of display stack then decrement linkcount and maxlinkcnt, thereby effectively remove top link
}
```

```
if(linkcount==maxlinkcnt){  
    hidecurrent();  
    linkcount--;  
}  
  
maxlinkcnt--; //lower max link count by 1  
  
loadcurrent();  
displaycurrent();  
loadnext();  
}//end removelink  
//-----  
  
function findtext(){  
    if(maxlinkcnt==0){return;} //Guard for open page  
    window.external.NavigateAndFind(links[linkcount],isubject.value,eIDArray[linkcount]);  
}//end findtext  
//-----  
  
function zoomup(){  
    if(maxlinkcnt==0){return;}  
    var zoomfactor = 1.0;  
    zoomfactor = parseFloat(document.getElementById(eIDArray[linkcount]).style.zoom);  
    zoomfactor = zoomfactor + 0.2;  
    if(zoomfactor > 10){return;}  
    document.getElementById(eIDArray[linkcount]).style.zoom = zoomfactor;  
}//end zoomup  
//-----  
  
function zoomdown(){  
    if(maxlinkcnt==0){return;}  
    var zoomfactor = 1.0;  
    zoomfactor = parseFloat(document.getElementById(eIDArray[linkcount]).style.zoom);  
    zoomfactor=zoomfactor - 0.2;
```

```
if(zoomfactor<0.1){return;}
document.getElementById(cIDArray[linkcount]).style.zoom = zoomfactor;
}//end zoomdown
//-----
function zoomwheel(){
    if(maxlinkcnt==0){return;}
    if(event.wheelDelta > 0){
        zoomup();
    }
    else{
        zoomdown();
    }
}//end zoomwheel
//-----
function onbeforeunloadpage() { //event handler is onbeforeunload
    //alert(window.event.type); //event doesn't bubble
    window.event.returnValue="To Stay in Search Engine, Please Hit Cancel Now."; //event is beforeunload
}
//-----
function onloadpage(){
    testing=false;
    attacheventhandlers();
    window.status="Loading Page, Please Wait....";
    playsound("loading.wav");
    initializeframes(); //initialize iFrames and hide
    document.getElementById(cIDArray[bottomofstack]).src="blank.html";
    document.getElementById(cIDArray[bottomofstack]).style.visibility='visible'; //make opening page visible
    document.getElementById(cIDArray[bottomofstack]).focus();
    if(isubject.value != ""){getnewlinks();} //if reloading and Subject is available, then re-establish links
}//end onloadpage
//-----
```

```
function onerrorpage(){
    playsound('explosion.wav');
    if(testing){alert("Error on Page");}
    //check??? window.onerror=null;
}//end onerrorpage
//-----

function onabortpage(){
    if(testing){alert("Abort triggered--reload initiated");}
    onloadpage();
}//end onerrorpage
//-----

function onunloadpage(){
    //alert("unloading event");
    playsound('goodbyc.wav');

    for(i=bottomofstack; i <= maxiframecount; i++){ //clear and hide all other frames
        document.getElementById(cIDArray[i]).style.visibility='hidden';
        document.getElementById(cIDArray[i]).src = "blank.html" //load blank page into frame to clear frame
    }
    document.getElementById(cIDArray[bottomofstack]).src="splash.html";
    document.getElementById(cIDArray[bottomofstack]).focus();
    document.getElementById(cIDArray[bottomofstack]).visibility='visible'; //make opening page visible
    detacheventhandlers();
    //Engine Resources automatically deallocated
}//end onunloadpage
//-----
```

  

```
function intervaltimerflop(){
    catch_setInterval = !catch_setInterval;

    if(catch_setInterval){
        document.getElementById('binterval').style.backgroundColor='rgb(0,255,0');
```

```
        }
    else{
        document.getElementById("binterval").style.backgroundColor='rgb(255,0,0)';
    }
}//end intervaltimerflop
//-----



function timeoutflop(){
    catch_setTimeout = !catch_setTimeout;

    if(catch_setTimeout){
        document.getElementById('btimeout').style.backgroundColor='rgb(0,255,0)';
    }
    else{
        document.getElementById('btimeout').style.backgroundColor='rgb(255,0,0)';
    }
}//end timeoutflop
//-----



function promptwindowflop(){
    catch_prompt = !catch_prompt;

    if(catch_prompt){
        document.getElementById('bprompt').style.backgroundColor='rgb(0,255,0)';
    }
    else{
        document.getElementById('bprompt').style.backgroundColor='rgb(255,0,0)';
    }
}//end promptwindowflop
//-----



function confirmwindowflop(){
    catch_confirm = !catch_confirm;

    if(catch_confirm){

```

```
        document.getElementById('bconfirm').style.backgroundColor='rgb(0,255,0)';
    }
} else{
    document.getElementById('bconfirm').style.backgroundColor='rgb(255,0,0)';
}
}//end confirmwindowflop
//-----



function openwindowflop(){
    catch_open = !catch_open;

    if(catch_open){
        document.getElementById('bopen').style.backgroundColor='rgb(0,255,0)';
    }
} else{
    document.getElementById('bopen').style.backgroundColor='rgb(255,0,0)';
}
}//end openwindowflop
//-----



function alertwindowflop(){
    catch_alert = !catch_alert;

    if(catch_alert){
        document.getElementById('balert').style.backgroundColor='rgb(0,255,0)';
    }
} else{
    document.getElementById('balert').style.backgroundColor='rgb(255,0,0)';
}
}//end alertwindowflop
//-----



function popupwindowflop(){
    catch_createPopup = !catch_createPopup;
```

```
if(catch_createPopup){
    document.getElementById('bpopup').style.backgroundColor='rgb(0,255,0)';
}
else{
    document.getElementById('bpopup').style.backgroundColor='rgb(255,0,0)';
}
}//end popupwindowflop
//-----
function helpwindowflop(){
    catch_showHelp = !catch_showHelp;

    if(catch_showHelp){
        document.getElementById('bhelp').style.backgroundColor='rgb(0,255,0)';
    }
    else{
        document.getElementById('bhelp').style.backgroundColor='rgb(255,0,0)';
    }
}//end helpwindowflop
//-----
function modalwindowflop(){
    catch_showModalDialog = !catch_showModalDialog;

    if(catch_showModalDialog){
        document.getElementById('bmodal').style.backgroundColor='rgb(0,255,0)';
    }
    else{
        document.getElementById('bmodal').style.backgroundColor='rgb(255,0,0)';
    }
}//end modalwindowflop
//-----
function modelesswindowflop(){
    catch_showModelessDialog = !catch_showModelessDialog;
```

```

if(catch_showModelessDialog){
    document.getElementById('bmodeless').style.backgroundColor='rgb(0,255,0)';
}
else{
    document.getElementById('bmodeless').style.backgroundColor='rgb(255,0,0)';
}
}//end modelesswindowflop
//-----
function colortrack(cid){
//catch_alert=false;
//alert(cid);
var xcolor = "";
var rsi=parseInt(document.getElementById(cid).style.backgroundColor.indexOf("("))+1;
var rei=parseInt(document.getElementById(cid).style.backgroundColor.indexOf(")"));
var rc=parseInt(document.getElementById(cid).style.backgroundColor.substring(rsi,rei));
var gsi=parseInt(document.getElementById(cid).style.backgroundColor.indexOf(","))+1;
var gei=parseInt(document.getElementById(cid).style.backgroundColor.lastIndexOf(","));
var gc=parseInt(document.getElementById(cid).style.backgroundColor.substring(gsi,gei));
if(rc==255 | gc==255){rc=255; gc=255;}
if(rc>=2 && gc>=2){rc=rc-2; gc=gc-2;}
xcolor = "rgb(" + rc + "," + gc + "," + "255)";
document.getElementById(cid).style.backgroundColor = xcolor;
return;
}//end colortrack
//-----
function setndisplay(){ //on GUI click
    if(maxlinkcnt==0){return;} //guard
    switch(parseInt(displaycnt)){
        case 1: displaycnt=2; break;
        case 2: displaycnt=4; break;
        case 4: displaycnt=1; break;
        default: displaycnt=1; break;
    }
}

```

```
//end switch

switch(displaycnt){
    case 1:
        for(i=bottomofstack; i <= maxiframecount; i++){
            document.getElementById(cIDArray[i]).style.visibility="hidden";
            document.getElementById(cIDArray[i]).style.left="0";
            document.getElementById(cIDArray[i]).style.top="4%";
            document.getElementById(cIDArray[i]).style.width="102%";
            document.getElementById(cIDArray[i]).style.height="96%";
        }
        break;
    case 2:
        for(i=bottomofstack; i<maxiframecount; i=i+2){
            document.getElementById(cIDArray[i]).style.visibility="hidden";
            document.getElementById(cIDArray[i]).style.left="0";
            document.getElementById(cIDArray[i]).style.top="4%";
            document.getElementById(cIDArray[i]).style.width="50%";
            document.getElementById(cIDArray[i]).style.height="96%";

            document.getElementById(cIDArray[i+1]).style.visibility="hidden";
            document.getElementById(cIDArray[i+1]).style.left="50%";
            document.getElementById(cIDArray[i+1]).style.top="4%";
            document.getElementById(cIDArray[i+1]).style.width="50%";
            document.getElementById(cIDArray[i+1]).style.height="96%";
        }
        break;
    case 4:
        for(i=bottomofstack; i<maxiframecount; i=i+4){
            document.getElementById(cIDArray[i]).style.visibility="hidden";
            document.getElementById(cIDArray[i]).style.left="0";
            document.getElementById(cIDArray[i]).style.top="4%";
            document.getElementById(cIDArray[i]).style.width="50%";
            document.getElementById(cIDArray[i]).style.height="48%";
        }
}
```

```

document.getElementById(cIDArray[i+1]).style.visibility="hidden";
document.getElementById(cIDArray[i+1]).style.left="50%";
document.getElementById(cIDArray[i+1]).style.top="4%";
document.getElementById(cIDArray[i+1]).style.width="50%";
document.getElementById(cIDArray[i+1]).style.height="48%";
}

for(i=bottomofstack+2; i<maxiframecount; i=i+4){
    document.getElementById(cIDArray[i]).style.visibility="hidden";
    document.getElementById(cIDArray[i]).style.left="0";
    document.getElementById(cIDArray[i]).style.top="52%";
    document.getElementById(cIDArray[i]).style.width="50%";
    document.getElementById(cIDArray[i]).style.height="48%";

    document.getElementById(cIDArray[i+1]).style.visibility="hidden";
    document.getElementById(cIDArray[i+1]).style.left="50%";
    document.getElementById(cIDArray[i+1]).style.top="52%";
    document.getElementById(cIDArray[i+1]).style.width="50%";
    document.getElementById(cIDArray[i+1]).style.height="48%";
}

break;
default:
for(i=bottomofstack; i <= maxiframecount; i++){
    document.getElementById(cIDArray[i]).style.left="0";
    document.getElementById(cIDArray[i]).style.top="4%";
    document.getElementById(cIDArray[i]).style.width="102%";
    document.getElementById(cIDArray[i]).style.height="96%";
}
break;
}//end switch

document.getElementById('bndisplay').value="D" + displayent;

for(linkcount=bottomofstack; linkcount<=displayent; linkcount++){
    displaycurrent();
}

```

```
    linkcount--; //linkcount set at last visible iFrame on GUI  
}//end ndisplay
```

```
//-----  
//-----  
//-----
```

```
//Test Functions
```

```
var cnt=0;  
function test(){  
return;
```

```
cnt++;  
catch_alert=false;
```

```
//if(cnt==1){  
    linkcount=1;  
    links[1]="testcatch.html";  
    loadcurrent();  
    displaycurrent();  
    return;  
}
```

```
return;
```

```
window.status="Test Functions";  
var rv=false;  
//set test state  
testing=true; //turn on alerts in event handlers  
linkcount=1;  
maxlinkcnt=1;  
catch_onblur=false;  
catch_onerrorupdate=false;  
catch_onerror=false;
```

```
catch_createPopup=false;
catch_open=false;
catch_confirm=false;
catch_setTimeout=false;
catch_setInterval=false;
catch_prompt=false;
catch_showHelp=false;
catch_showModalDialog=false;
catch_showModelessDialog=false;
catch_alert=false; //important to turn off alert check checking when testing
```

```
//-----
```

```
//showModalDialog test
```

```
catch_showModalDialog=true; //set checking on
alert("catch showModalDialog true");
var rtnx = showModalDialog("modaltest.htm","my arguments passed","dialogWidth:300px;dialogHeight:300px"); //triggers showModalDialog event
alert("Return Value from showModalDialog catcher: " + rtnx);
```

```
catch_showModalDialog=false;
alert("catch showModalDialog false");
var rtny = showModalDialog("modaltest.htm","my arguments passed","dialogWidth:300px;dialogHeight:300px"); //triggers showModalDialog event
//if X out, returns null
//cancel returns "" empty
//OK returns string; if string is empty, returns empty ""
alert("Return Value from showModalDialog catcher: " + rtny);
```

```
//-----
```

```
//showModelessDialog test
```

```
catch_showModelessDialog=true; //set checking on
alert("catch showModelessDialog true");
var rtnz = showModelessDialog("modaltest.htm","my arguments passed","dialogWidth:300px;dialogHeight:300px"); //triggers showModelessDialog
event
alert("Return Value from showModelessDialog catcher: " + rtnz);
```

```
catch_showModelessDialog=false;
alert("catch showModelessDialog false");
var rtna = showModelessDialog("modaltest.htm","my arguments passed","dialogWidth:300px;dialogHeight:300px"); //triggers showModelessDialog
event
//if X out, returns null
//cancel returns "" empty
//OK returns string; if string is empty, returns empty ""
alert("Return Value from showModelessDialog catcher: " + rtna.location.href);

//-----
//showhelp test

catch_showHelp=true; //set checking on
alert("catch showHelp true");
window.showHelp("testshowhelp.chm");

catch_showHelp=false; //set checking off
alert("catch showHelp false");
window.showHelp("testshowhelp.chm");

//-----
//onblur test

catch_onblur=true;
alert(document.getElementById(eIDArray[linkcount]).id + " catch onblur true:");
rv=document.getElementById(eIDArray[linkcount]).fireEvent("onblur");
//onblur should catch and reset focus, so blur again to check if focus has been reset
rv=document.getElementById(eIDArray[linkcount]).fireEvent("onblur");

catch_onblur=false;
alert(document.getElementById(eIDArray[linkcount]).id + " catch onblur false:");
rv=document.getElementById(eIDArray[linkcount]).fireEvent("onblur");
//should have same result second time
rv=document.getElementById(eIDArray[linkcount]).fireEvent("onblur");
```

```
//-----
//onerrorupdate test

catch_onerrorupdate=true;
alert(document.getElementById(eIDArray[linkcount]).id + " catch onerrorupdate true.");
rv=document.getElementById(eIDArray[linkcount]).fireEvent("onerrorupdate");
if(rv != true){Alert("fireEvent onerrorupdate failed");}
//should have same result second time
rv=document.getElementById(eIDArray[linkcount]).fireEvent("onerrorupdate");
if(rv != true){Alert("fireEvent onerrorupdate failed");}

catch_onerrorupdate=false;
alert(document.getElementById(eIDArray[linkcount]).id + " catch onerrorupdate false.");
rv=document.getElementById(eIDArray[linkcount]).fireEvent("onerrorupdate");
if(rv != true){Alert("fireEvent onerrorupdate failed");}
//should have same result second time
rv=document.getElementById(eIDArray[linkcount]).fireEvent("onerrorupdate");
if(rv != true){Alert("fireEvent onerrorupdate failed");}

//-----
//alert test

catch_alert=true; //set checking on
window.status="catch alert true";
window.alert("You shouldn't see this if method trapped!"); //triggers alert event; no return value

catch_alert=false; //set checking off
window.alert("catch alert false"); //triggers alert event

//-----
//confirm test

catch_confirm=true; //set checking on
alert("catch confirm true");
var rtn = window.confirm("You shouldn't see this if method trapped!"); //triggers confirm event
```

```
alert("Return Value from confirm catcher: " + rtn);

catch_confirm=false;
alert("catch confirm false");
rtn = window.confirm("Please confirm this method test"); //triggers confirm event
alert("Return Value from confirm catcher: " + rtn);
//-----
//prompt test

catch_prompt=true; //set checking on
alert("catch prompt true");
var rtn = window.prompt("You shouldn't see this if method trapped !","Prompt not Caught"); //triggers prompt event
alert("Return Value from prompt catcher: " + rtn);

catch_prompt=false;
alert("catch prompt false");
rtn = window.prompt("prompt method test"); //triggers prompt event
alert("Return Value from prompt catcher: " + rtn);
//-----
//open test

catch_open=true; //set checking on
alert("catch open true");
var rtn = window.open("blank.html",null,"height=400,width=400,status=yes,toolbar=yes,menubar=yes,location=yes"); //triggers open event
alert("Return Value from open catcher: " + rtn);
catch_open=false;
alert("catch open false");
rtn = window.open("blank.html","newwindowtarget","height=400,width=400,status=yes,toolbar=yes,menubar=yes,location=yes",true); //triggers open
event
alert("Return Value from open catcher: " + rtn.location.href);

//-----
//setTimeout test

catch_setTimeout=true; //set checking on
```

```
alert("catch setTimeout true");
rtn = window.setTimeout("status='setTimout catch true'",2500,"JavaScript"); //triggers setTimeout event
alert("Return Value from setTimeout catcher: " + rtn);
catch_setTimeout=false; //set checking off
alert("catch setTimeout false");
rtn = window.setTimeout("status='setTimout catch false'",2500,"JavaScript"); //triggers setTimeout event
alert("Return Value from setTimeout catcher: " + rtn);
```

//-----

```
//setInterval test

catch_setInterval=true; //set checking on
alert("catch setInterval true");
rtn = window.setInterval("status='setInterval catch true'",5000,"JavaScript"); //triggers setInterval event
alert("Return Value from setInterval catcher: " + rtn);
catch_setInterval=false; //set checking off
alert("catch setInterval false");
rtn = window.setInterval("status='setInterval catch false'",5000,"JavaScript"); //triggers setInterval event
alert("Return Value from setInterval catcher: " + rtn);
```

//-----

```
//createPopup test

catch_createPopup=true;
alert("catch createPopup true");
var Pop1 = window.createPopup(); //triggers createPopup event
alert("Return Value from createPopup catcher: " + Pop1);
//popWindow(Pop1); //make popup window: will throw error as Pop1 doesn't exist

catch_createPopup=false;
alert("catch createPopup false");
var Pop2 = window.createPopup(); //triggers createPopup event
alert("Return Value from createPopup catcher: " + Pop2);
popWindow(Pop2); //make popup window
```

```
//-----
//leave in production state
testing=false;
catch_onblur=true;
catch_onerrorupdate=true;
catch_onerror=true;
catch_alert=true;
catch_createPopup=true;
catch_open=true;
catch_confirm=true;
catch_setTimeout=true;
catch_setInterval=true;
catch_showHelp=true;
catch_showModalDialog=true;
catch_showModelessDialog=true;
catch_prompt=true;
}
//-----
```

```
function popWindow(pop){
    var wPopBody = pop.document.body;
    wPopBody.style.backgroundColor = "lightyellow";
    wPopBody.style.border = "solid black 1px";
    wPopBody.innerHTML = "Click outside <B>Popup</B> to close window.";
    pop.show(100, 100, 400, 400, document.body);
    return true;
}
//-----
```

Contents of file engine.java:

```
import java.io.*;
import java.net.*;
import java.awt.*;
import java.applet.*;
import netscape.javascript.JSObject;
import netscape.javascript.JSEException;

public class engine extends Applet{
    String usersubject="znone";
    int linkgetcount=0;
    String LinkSetResponses[] = new String[10];
    int LinkSetResponseCount = 0;
    boolean debug = true;

//-----
public void init() {
    System.out.println("initializing... ");
}

//-----
public void start() {
    System.out.println("auto start... ");
}

//-----
public void stop() {
    System.out.println("deallocating resources... ");
    System.runFinalization(); //deallocated resources
    System.out.println("garbage collection... ");
    System.gc(); //garbage collection
}

//-----
```

```
public void destroy() {
    System.out.println("unloading...");
}

//-----
public void setssubject(String newssubject) { //called from javascript to set the subject
    userssubject = newssubject;
    System.out.println("Setting subject: " + userssubject);
}

//-----
public void setlinkgetcount(String newlinkcount) { //called from javascript to set the number of links returned
int nlc = Integer.parseInt(newlinkcount);

    //System.out.println("Setting nlc: " + nlc);
    linkgetcount = nlc;
    if(debug){System.out.println("Setting link get count: " + linkgetcount);} //10 is granularity
}

//-----
public void getlinkset(int whichset) {
String resource, host, file, line, subject="fractals";
String responses[] = new String[10];
int slashPos=0, blankPos=0, gtPos=0, responseCount=0;

try{
    if(userssubject.equals("znone")) //no user subject set yet
        return;
    else
        subject=userssubject; //set from javascript call

    for (int j=0; j<=subject.length()-1; j++){
        if(subject.substring(j,j+1).equals(" ")){
            //System.out.println(subject.substring(0,j));

```

```
//System.out.println(subject.substring(j,j+1));
//System.out.println(subject.substring(j+1,subject.length()));
subject = subject.substring(0,j) + "%20" + subject.substring(j+1,subject.length());
if(debug){System.out.println(subject);}
}
}

resource="http://www.google.com/search?q=" + subject + "&hl=en&lr=&start=" + whichset + "&sa=N";
if(debug){System.out.println(resource);}

resource = resource.substring(7); // skip http://

slashPos = resource.indexOf('/'); // find host/file separator
file = resource.substring(slashPos); // isolate host and file parts
host = resource.substring(0,slashPos);

http webConnection = new http(host,debug);

if(webConnection != null) {
    BufferedReader bisr = webConnection.get(file,debug);

    while( (line = bisr.readLine()) != null) { // read until end of stream
        if(line.length() >= 18 && line.substring(0,18).equals("<p><a href=http://>")){
            gtPos = line.substring(11).indexOf(">");
            responses[responseCount]=line.substring(11,gtPos+11);
            responseCount++;
        }
    }
}

//set globals
//if(debug){System.out.println("responseCount: "+ responseCount);}
LinkSetResponseCount=responseCount;
LinkSetResponses=responses;
return;
```

```
}//end try

catch (Exception e) {
    System.runFinalization(); //deallocated resources
    System.gc(); //garbage collection
} //end catch

}//end getlinkset

//-----
public void enginestart() throws JSException {
JSObject win = JSObject.getWindow(this);
JSObject doc = (JSObject) win.getMember("document");
JSObject loc = (JSObject) doc.getMember("location");
String localhref = (String) loc.getMember("href"); // document.location.href
//JSObject.getWindow (this).eval ("javascript:alert('This is a test of the National Broadcast System!')");
String tempstr;
String maxlinkcount[] = new String[1];
String fsrc[] = new String[1];
String AggregateLinkResponses[] = new String[50];
int AggregateLinkCount = 0;
int lastindex = 0;
int newlastindex = 0;

try{
    //if(debug){System.out.println("Requesting Links:" + linkgetcount);}
    if(linkgetcount>50){
        System.out.println("Requested Link Count too High--50 max: " + linkgetcount);
        return;
    }

    for (int j=0; j<=linkgetcount-10; j=j+10){ //10,20,30,40,50
        getlinkset(j);
    }
}
```

```

//keep track of total links found on successive calls to getlinkset()

AggregateLinkCount = AggregateLinkCount + LinkSetResponseCount;

//if(debug){System.out.println("AggregateLinkCount: " + AggregateLinkCount);}
//if(debug){System.out.println("LinkSetResponseCount: " + LinkSetResponseCount);}
//if(debug){System.out.println("lastindex: " + lastindex);}
//if(debug){System.out.println("loop finish: " + (lastindex + LinkSetResponseCount - 1));}

for (int i=lastindex; i <= (lastindex + LinkSetResponseCount - 1); i++){
    //if(debug){System.out.println("LSR: " + LinkSetResponses[i-lastindex]);}
    AggregateLinkResponses[i]=LinkSetResponses[i-lastindex]; //index starts at 0; LinkSetResponses index only goes from 0-9 max
    newlastindex++;
} //end inside for

lastindex = newlastindex;
//if(debug){System.out.println("newlastindex: " + lastindex);}
} //end outside for

tempstr="alan" + AggregateLinkCount; //convert int to string
tempstr=tempstr.substring(4);
maxlinkcount[0]=tempstr;
if(debug){System.out.println("Setting maxlinkcount in JS: " + tempstr);}
win.call("setlinkcount", maxlinkcount); //Send AggregateLinkCount to javascript

for (int k=0; k<=AggregateLinkCount-1; k++){
    if(debug){System.out.println(AggregateLinkResponses[k]);}
    fsrc[0] = AggregateLinkResponses[k];
    win.call("linker", fsrc);
} //end for

if(debug){System.out.println("\nDone.\n");}
} //end try

catch (Exception e) {
    System.runFinalization(); //deallocated resources
}

```

```
System.gc(); //garbage collection
} //end catch

}//end enginestart
}//End Applet

//-----
//-----
class http {

InetAddress WWWhost;

public http (String host, boolean debug) throws UnknownHostException {
    WWWhost = InetAddress.getByName(host); //Set WWWhost
    if(debug){System.out.println("WWW host = " + WWWhost);}
}

public BufferedReader get (String file,boolean debug) throws IOException { //Returns BufferedReader
    int HTTP_PORT = 80;
    String response;

    Socket httpPipe = new Socket(WWWhost,HTTP_PORT);
    if((httpPipe == null) {return null; }

    InputStream inn = httpPipe.getInputStream();      // get raw streams
    InputStreamReader isr = new InputStreamReader(inn); //convert bytes to Unicode characters
    BufferedReader bisr = new BufferedReader(isr);   //buffer data to enhance performance

    OutputStream outt = httpPipe.getOutputStream();
    PrintStream out = new PrintStream(outt);      //convert to character output- 8 bit ASCII representation
    //Don't use PrintWriter, it converts to 16 bit bytecode representation

    if((inn==null || outt==null) {
        System.out.println("Failed to open streams to socket.");
        return null;
    }
}
```

```
    }

// send GET request
out.println("GET " + file + " HTTP/1.0\n");
//if(debug){System.out.println("GET " + file + " HTTP/1.0\n");}

// read response until blank separator line
while ( (response = bisr.readLine()).length() > 0 ) {
    if(debug){System.out.println(response);}
}

return bisr; // return BufferedReader to allow client to read resource
}//End InputStream
}//End http
```